A General Purpose Software Evaluation System

Behrouz H. Far and Vani Mudigonda
Schulich School of Engineering, University of Calgary, Calgary, Alberta, Canada
E-mail: far@ucalgary.ca, vmudigon@ucalgary.ca

Abdel-Halim Elamy
Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada
E-mail: elamy@ualberta.ca

Keywords: SW evaluation, statistics.

Received: October 12, 2008

In the present day market situation, there are several alternatives available when a customer wants to purchase a product or adopt a software system that meets the customer's requirements. General Purpose Software Evaluation (GPSE) system uses state of the art statistical methods based on Multidimensional Weighted Attribute Framework (MWAF) for the evaluation of the available alternatives. By using GPSE system, the user can follow the MWAF process and design the architecture which best describes the given evaluation problem. The architectural elements of MWAF essentially focus on survey questionnaire which involves gathering information from several domain experts. The GPSE system then applies principles of Analysis of Variance (ANOVA) and Tukey's pairwise comparison tests on the data collected to arrive at selection of the best suited alternative for the given problem. The GPSE system has been fully implemented and successfully tested on several projects including evaluation of multi-agent development methodologies and selection of COTS products.

Povzetek: Predstavljen je splošni sistem GPSE za vrednostenje programskih sistemov..

1 Introduction

Software technologies have been evolving rapidly and for a given set of functional and non-functional requirements there usually exist several competing software products. The present day users are faced with a challenging situation that requires evaluation and selection of a suitable software product that satisfies the users' operational and business needs. Unfortunately this evaluation is usually carried on in an ad-hoc and informal way and with various degree of success. The objective of this research is to develop a General Purpose Software Evaluation (GPSE) system that helps a user systematically evaluate a set of alternative products available for a given set of requirements by employing sound statistical methods. The GPSE system incorporates and implements the Multidimensional Weighted Attribute Framework (MWAF) [10, 11]. MWAF is a framework for creating the evaluation criteria and collecting data from subject matter experts in the form of rates and weights for each alternative included in the evaluation. The data collected is then processed and subjected to statistical analysis by using Analysis of Variance (ANOVA) and Tukey's pairwise comparison tests. The MWAF possesses great potential in its applicability to a variety of applications. The present work focused on implementing the MWAF in the form of the GPSE system.

The paper is organized as follows. Section 2 delivers a quick review of several software evaluation techniques. Section 3 presents overview of the MWAF framework and Section 4 describes the GPSE system design and implementation details. Finally, Section 5 presents the conclusions and scope for future extension of the project.

2 Related works

Software evaluation methodologies can be divided into two categories. The first category is used to evaluate software development methodologies or processes such as those used to evaluate various agent-based development methodologies. The second category is used to evaluate software products such as COTS evaluation and selection methodologies.

In the literature, there are a few studies addressing the comparison and evaluation of processes and methodologies. Available techniques merely focus on a single application domain making generalization of the method almost impossible. For instance in the domain of evaluating agent-based development methodologies, Dam et al [9] proposed an attribute-based framework for evaluation by analyzing feedback data from both the system developers as well as from end users. Juneidi and Vouros [14] utilized the evaluation criteria of Shehory and Sturm [21] and conducted a study to evaluate three agent-based development methodologies. Further, Bayer and Svantesson [2] introduced a study to compare and evaluate two agent-based methodologies by identifying their strengths and weaknesses. Another work has been presented by Sudeikat et al. [23] to evaluate three agent-based methodologies (MaSE, Tropos, and Prometheus)

against a number of evaluation criteria (e.g., internal architecture, social architecture, communication, and process-related features) that have been examined and compared qualitatively.

Tran et al. present a comparative Feature Analysis Framework [24] that includes 4 criteria: process, technical, model and support, and is tailored to evaluating agent-based methodologies. The Framework can be recommended to adopt as an analytical tool to exhibit various detailed features involving agents and multi-agent systems. Yet, it is not a purely evaluation framework.

Silva et al. proposed a Non-Functional Requirements (NFR) framework to describe the internal properties of systems and to evaluate the agent-based methodologies based on these properties [20]. As a matter of fact, non-functional requirements (NFRs) have significant impact on the process of software development [7]. When designing a system, such NFRs represent trade-offs in the design basic principles that contribute to deciding upon specific structural/behavioral aspects of the system [13]. Similar to Tran et al.'s framework, Silva et al.'s is lacking the empirical/analytical approach to quantify the subjective features of the NFRs which are qualitative in their nature and consequently, cannot be easily and accurately examined and compared.

Regarding product evaluation methodologies, there are relatively larger number of methods such as those used to evaluate and select commercial off-the-shelf (COTS) products [1, 3, 4, 7, 15, 16, 18].

To conclude, all of the above mentioned techniques show one or more of the following methodological deficiencies:

a) unrepresentative set of responses;
b) heterogeneous experimental subjects;
c) using different instruments for obtaining similar responses; and
d) mixing up the scales of measurement.

Our GPSE system is generic enough to be used for evaluation of both software products and processes. The Multidimensional Weighted Attribute Framework (MWAF), which is used in the GPSE system, follows sound statistical guidelines to design experiments and interpret data and consequently does not suffer from the above mentioned deficiencies.

3 Multidimensional Weighted Attribute Framework (MWAF)

In this section we present the Multidimensional Weighted-Attributes Framework (MWAF) for software system evaluation. MWAF is a general-purpose framework that can be adapted to evaluate software products, e.g., programming languages, operating systems, software engineering methodologies, software development toolkits and software communications protocols.

3.1 MWAF Framework

The main idea of MWAF is to define the most common and important criteria (or dimensions) of the system being evaluated, identifying the attributes that describe each of these dimensions, and then evaluating each dimension through its attributes against all the potential systems that are selected for evaluation. As shown in Figure 1, MWAF consists of the following three main components:

1) *Dimensions:* the framework comprises a number of dimensions, each of which represents one of the major evaluation criteria.
2) *Attributes:* are the different features pertaining to each criterion (i.e. dimension) to describe it using a set of definite questions.
3) *Parameters:* the numeric values that are given to measure the attributes.

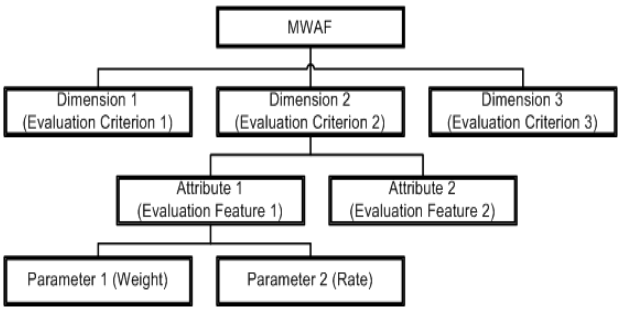

Figure 1: Hierarchy of the Multidimensional Weighted-Attributes Framework (MWAF).

For example, the following four attributes can be used for evaluating the 'objectivity' of public websites: a) Goal-orientation; b) Comprehensiveness; c) Fair-mindedness; and d) Independency. To perform this evaluation, we can assume 'objectivity' as an evaluation dimension that encompasses the above 4 attributes. Each of these attributes can be evaluated through relevant expressive questions, such as:

a. Goal-orientation: to what degree does the website meet its announced goals?
b. Comprehensiveness: how detailed is the information posted on the website?
c. Fair-mindedness: to what extent would you agree with the opinions expressed by the authors of the website?
d. Independency: to what degree would you reject the popped up advertising on this web page?

When applying MWAF, several expert users will be asked to give two parameters to each of the evaluated attributes: a *weight* to identify the importance of the attribute, and a *rate* to measure its strength or effectiveness. Weight is a subjective parameter, as it entirely relies upon the evaluator's personal opinion. On the other hand, rate is an objective parameter because it is measured according to the degree of availability or effectiveness of the examined property as represented by the evaluated attribute. In MWAF, the values given to the two parameters are numeric and range from 0 to 10. A value of '0' implies full absence of the measured

attribute, whereas a value of 10 reflects its maximum availability and strength. For instance, the '*Comprehensiveness'* attribute may receive weights and rates by four participants as shown in Table 1.

Table 1: Sample expert user input.

Expert User	1	2	3	4
Weight	5	9	6	7
Rate	10	7	8	9

In this example, the first expert user assumes that the '*Comprehensiveness'* is moderately important to evaluate a public website. However, in his/her view, the evaluated website is extremely comprehensive. Based on the collected data, we can determine the weighted rates by normalizing each raw rate against the average weight given to this attribute.

$$Average\ Weight = \frac{\sum w_i}{n} = \frac{5+9+6+7}{4} = 6.75$$

And the weighted rates are depicted in Table 2.

Table 2: Calculated weighted rates.

Expert User	1	2	3	4
Weighted Rate	6.750	4.725	5.400	6.075

The rest of the evaluating procedure will be carried out upon analyzing and comparing these rates, as weighted against the average importance of the evaluated attribute.

In order to take a broad view of the final conclusions and findings, each system shall be evaluated by several expert users. The number of expert users will be identified during the experiment design (See Section 3.3 step 5).

3.2 MWAF Data Abstraction

The data abstraction process formulates blocks and replicas based on the identified dimensions and attributes (see Figure 2). A block consists of a set of treatments assigned to an expert user for evaluation. Each treatment is included in multiple blocks and hence evaluated by multiple expert users leading to multiple replicas of data. Identification of blocks and replica is part of the experiment design. For example, in the Balanced Incomplete Block Design (BIBD) model each pair of treatment must occur the same number of times as any other pair. The common choices are Completely Randomized Design (CRD), Randomized Complete Block Design (RCBD), and Balanced Incomplete Block Design (BIBD) model.

The CRD is the simplest type of randomization schemes in that subjects are assigned to treatments completely at random such that every experimental unit has an equal chance to receive any of the available treatments [19]. Various randomization techniques could be used for assigning subjects to treatment groups; the common method is to label subjects or treatments and then use a table of random numbers to select subjects at random and assign them to treatments.

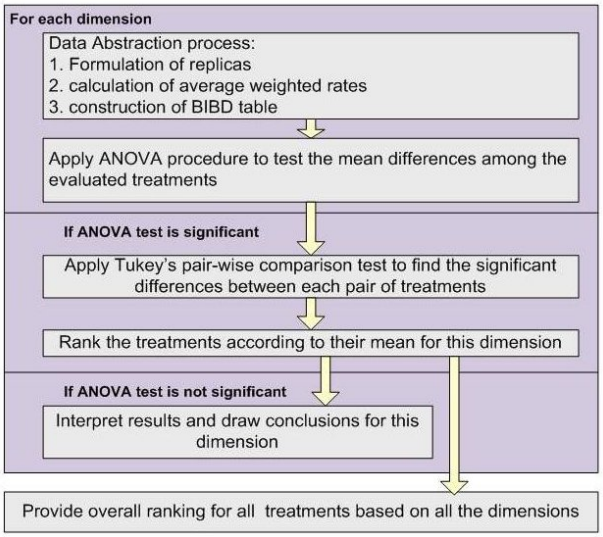

Figure 2: MWAF process flow diagram.

Cochran and Cox [6] indicate that due to the unrestricted randomization, units that receive one treatment may be naturally different from units that receive other treatments. This heterogeneity among experimental units contributes to producing a larger experimental error as compared to other designs. However, for the same number of observations a completely randomized design has the largest degrees of freedom for error. Although the sum of squares error may be enlarged by the natural variability in units, dividing this sum of squares by larger degrees of freedom may result smaller mean square error. Given the above advantages and disadvantages, CRD is appropriate when experimental units are quite homogenous, the experiment under study is relatively small, and when there is a chance to lose some experimental units and having a missing data problem.

In RCBD, randomization is restricted and controlled such that the experimental units are arranged into homogeneous groups (called blocks), and the treatments are then assigned at random to these blocks so that each treatment occurs once in every block, or as planned if the block sizes are not the same. The rationale behind blocking is to minimize the variability among units within blocks while maximizing it among blocks. Neter argues that RCBD can potentially have disadvantages such as: more assumptions (e.g., no interactions between treatments and blocks, and constant variability among blocks) are needed to be met; missing observations are complex to handle; and precision decreases as the number of experimental units in a block increases [19].

Clarke and Kempson [5] indicate that experiments often use supplies or resources that are not homogenous, but can be arranged into blocks of similar units so that most of the heterogeneity is taken out between blocks. An incomplete block design is called "balanced" or "symmetrical" if treatment levels are binary [17]. That is, when an incomplete design is formed so that every pair of treatments occurs together the same number of times as any other pair, the design is a Balanced Incomplete Block Design (BIBD). In BIBD, all

treatment comparisons are of the same accuracy, thus, we use these designs when all treatments are equally important. Yates [25] argues that the main drawback of a BIBD is that the number of replications required is in most cases large when the number of treatments is at all large. However, we can overcome this drawback by administering with the condition of balance, but at the cost of some loss of efficiency in addition to the inconvenience of having slight variation in accuracy for different sets of treatment comparisons.

Experience shows that the BIBD is usually the appropriate design to adopt in evaluating software systems. The reason is that, evaluating software systems is usually not constrained by using sensitive resources, the situation that may limit many biological or chemical experiments from being conducted. In fact, the resources needed for software evaluation (e.g., software products and expert users) are usually manageable, or at least can be controlled at the expense of having more expert users.

3.3 MWAF Process

The MWAF is an eight step process as defined below (see Figure 2):

Step 1. Select target software products

To select the software products being evaluated, one starts with conducting a primarily survey to review a set of competing candidates and select the most qualified ones. A qualified product can be defined as the one that satisfies some generic assumptions such as: (a) has reasonable documentation to describe it; (b) is fairly known to the community; and (c) has a reasonable domain of applicability, etc.

Step 2. Identify dimensions

In this step, one identifies a set of the evaluation criteria that represent the dimensions and the hierarchal structure of our framework. Examples of the dimensions are: modeling, communication, process, support, etc.

Step 3. Identify attributes

In this step, one determines the relevant features (i.e., attributes) pertained to each dimension. This also includes constructing a hierarchy structure and validating its consistency to ensure that no redundancies exist among the attributes for all the dimensions. For example, modeling-related dimension may consist of attributes that address and examine the most common and important aspects to model the product, such as: notation, expressiveness, abstraction, consistency, concurrency, traceability, derivation, reusability, etc.

Step 4. Design questionnaire

One has to design a set of questions corresponding to the dimensions and their attributes. The questions must be understandable, unambiguous, and provide clear statements to examine the effectiveness and strength of the related attributes. When designing questionnaire, it is important to set up the appropriate scale of measurement (e.g., nominal, ordinal, interval, or ratio) based on the nature of the collected data [22].

Step 5. Select statistical model

To perform analysis, one has to select the most appropriate statistical model and procedure that can fit and treat the data. This step is also helpful to determine the proper number of observations needed (and consequently the number of expert users needed to give their feedback to the evaluation questionnaire) to achieve reasonable accuracy of the statistical analysis.

Step 6. Select expert users

After determining the proper number of expert users (aka. participants), one has to select qualified participants to deliver the questionnaire with detailed guidelines to assure clarity. It is also recommended to hold instructional sessions to explain the evaluation task, the anticipated results, and the proper way to respond to the questionnaire. The participants should receive sufficient documentation about the products being evaluated, clear instructions about the experiment, and equal amount of time to complete their tasks.

Step 7. Collect and validate responses

The collected data will be validated to assure completeness and accuracy. One way to do this is to simply run a rough test on the collected data to detect outliers, for instance, by using scatter plots. In the case that outliers are observed, it is recommended to consult with the expert users who provided the data to make sure that the meant values are correct and not mistakenly recorded.

Step 8. Perform statistical analysis

The major step for implementing MWAF is to conduct a statistical experiment to evaluate the given products (see Figure 2). Prior to this step, one has to identify the statistical hypotheses and end up by testing the statistical significance of the hypotheses, analyzing the obtained results and drawing the final conclusions. The statistical hypotheses are:

- *Null hypothesis:* There is no significant difference in the mean effectiveness of the examined dimension among the evaluated products.
- *Alternative hypothesis:* There is a significant difference in the mean effectiveness.

Then, one can analyze the data statistically by applying the analysis of variances (ANOVA) procedure to the model. The underlying idea of ANOVA is to compare the variability of the observations between groups to the variability within groups. If the variability between groups is smaller than the variability within groups, it means that different groups are not significantly different, whereas if the variability between groups is larger than the variability within groups, it implies that different groups are significantly different.

If some variability is identified among the evaluated products on a certain dimension, Tukey's test for pair-wise comparison of the products is performed to test for multiple comparisons to identify which products are actually different. In contrast, if the overall ANOVA test was insignificant, applying any pairwise comparison is not necessary. In such a case, the conclusion to be made

is that all the products are statistically equal in their main effects against the attributes of the examined dimension.

The treatments are then ranked based on their means calculated for each dimension. An overall ranking of the treatments is finally calculated statistically.

In should be noted that prior to applying ANOVA, one may elect to examine the adequacy assumptions by testing the outliers, normality of residuals, and the homogeneity of residuals [10]. Table 1 shows the tests defined and used in the GPSE system.

Table 3: Suggested ANOVA tests.

Test	Test Type	Instrument Used
1	Outliers	a. Normal probability plot of residuals b. Individual value plot of residuals versus independent variable
2	Normality of residuals	Normal probability plot of residuals
3	Homogeneity of error variances	a. Residual plots against fitted values b. Bartlett's test

3.4 MWAF Advantages

Compatibility: MWAF is capable to conduct evaluation studies that are similar to many cases presented in the literature. This is because MWAF recognizes and integrates the important features of other frameworks, overcomes any obvious deficiencies, and adopts new features that generalize and extend its usability.

Structure: MWAF can be represented by an effective hierarchical structure, which derives its power from the principle of 'divide and conquer' that contributes to analyzing a complete taxonomy of evaluation attributes.

Scalability: MWAF is flexible to scaling up/down in order to expand or reduce its dimensions and/or attributes. In addition to its capability in supporting the conventional evaluation of software, MWAF can fit evaluation studies that are characterized by its dynamic nature; for instance, optimizing an objective function (e.g., maximizing overall performance, marketability, or minimizing costs or potential risks) by simulating potential features that can be released to a new product.

4 GPSE system analysis and design

The current stand-alone implementation of the GPSE system has a graphical user interface (GUI), a database, and a statistical analysis unit as shown in Figure 3.

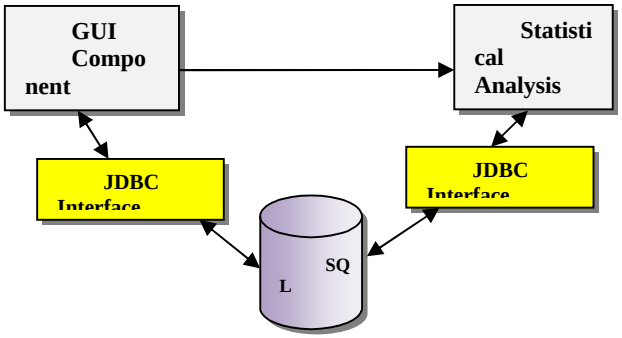

Figure 3: Overall architecture of GPSE system.

The functionalities of GUI facilitate configuration of the MWAF framework, collection of data comprising expert users' ratings, initiation of the statistical evaluation process, and displaying of the analysis results. A database is required for storing and retrieving information pertaining to the MWAF configuration data, expert users' ratings, and results of statistical analysis. As explained in Section 3, the key functionalities of the analysis unit include ANOVA method and Tukey's pairwise comparison tests. Figure 4 depicts the interactions among the system components.

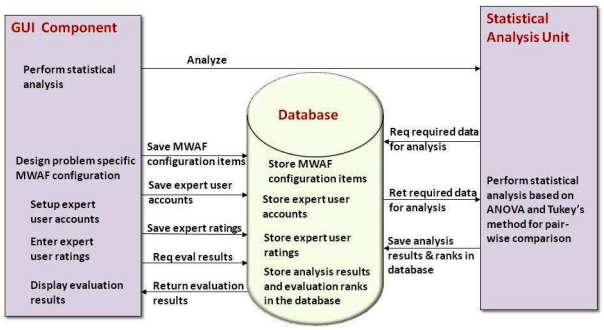

Figure 4: GPSE system component interaction.

The GPSE system is implemented using JAVA technologies (http://java.sun.com). MySQL database (http://www.mysql.com/) is used for data storage and retrieval purposes. In order to access MySQL database, Java Database Connectivity (JDBC) is used from both GUI component and statistical unit.

Current version of the GPSE system can be broken down into six main functionalities or Use Case Diagrams (UCD) as shown in Figure 5.

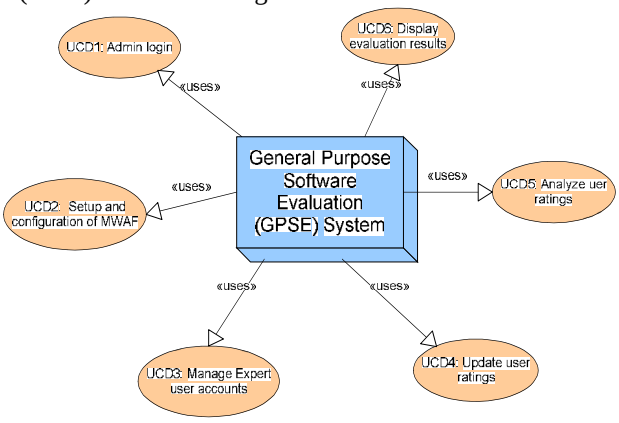

Figure 5: High level functionality diagram representing the functional decomposition of the GPSE system.

UCD1 – Admin Login: This functionality deals with logging into the GPSE system with administrator privileges. Administrator logs into the GPSE system in order to create expert users' accounts, setup the MWAF configuration framework, initiate the statistical analysis on the ratings data provided by expert users, and display results for the software evaluation problem.

UCD2 – Setup and Configuration of MWAF: This functionality deals with the configuration of MWAF architecture for the given evaluation problem. As part of this configuration setup the administrator will use the

GPSE GUI to setup the names of dimensions and the attributes for each dimension for various products that need to be evaluated. Subsequently, the framework will be accessed by various experts to provide ratings. Along with dimensions, attributes, and treatments, the MWAF framework also consists of blocks and replicas. A block consists of a set of treatments. Each user is assigned with a block of treatments to be evaluated. Managing blocks in terms of addition and deletion of blocks as well as assigning treatments to blocks is dealt by this functionality. A replica consists of the whole set of treatments considered for the particular evaluation problem. Managing the number of replicas for this evaluation problem is also included in this functionality.

UCD3 – Manage Expert User Accounts: This functionality deals with the creation of user accounts for various expert users who will use the system in order to provide their expert ratings based on the framework designed by the Administrator. During the creation of expert users of the system, each expert user is assigned with a block of treatments in order to provide their ratings to the assigned block of treatments.

UCD4 – Update User Ratings: The functionality covers the aspects related to the expert user log in, updating of ratings provided by the expert user, and saving the expert ratings to the database.

UCD5 – Analyze User Ratings: This functionality captures the statistical analysis of the evaluation process. After all the user ratings are provided, the administrator will access the system to initiate statistical analysis of the data stored in the system for the given evaluation problem. The administrator will request the system to analyze the user ratings data.

UCD6 – Display Evaluation Results: This functionality captures the aspects related to the display of the evaluation results. After the ratings from various expert users are analyzed, the administrator instructs the system to generate output results for display. Using this functionality, the administrator can also provide and save summary and recommendations for the evaluation conducted.

The stand-alone GPSE system is fully implemented, tested and verified. Figure 4 shows various screenshots of the stand-alone GPSE system. The system has been tested rigorously with the data that was collected in a set of experiments [10, 11] for the evaluation of various agent-oriented software engineering (AOSE) methodologies. The results obtained from the manual data analysis in the experiments were compared with the results from the GPSE system. Hence, the system fulfils its technical goals in that a functional GPSE system based on MWAF is developed and meets the desired objectives.

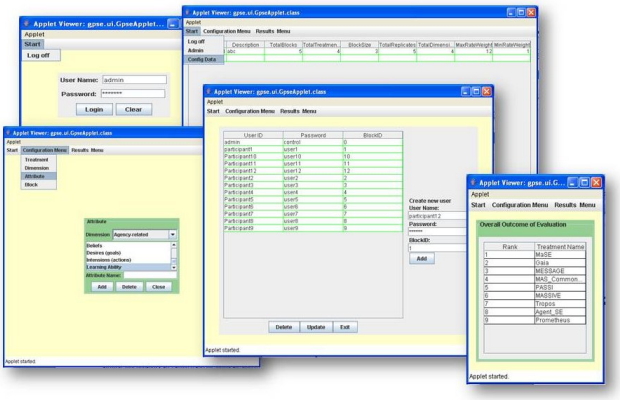

Figure 6: Screen shots of the GPSE systems.

Currently the GPSE system is being redesigned and converted to a multi-agent system application. In this case, several remote expert users may provide their ratings for the products that need to be evaluated. Furthermore, the MWAF process is also being automated using a multi-agent system in which intelligent software agents are responsible for steps of the MWAF process. For example, an agent, incorporated with the knowledge of designing statistical experiments, is responsible for the design of experiment (Step 5 of the MWAF process) and another, incorporated with data mining capabilities, for selecting dimensions and attributes (Steps 2-3 of the MWAF process). These dimensions and attributes are then sent to other agents that act as personal assistant agents for expert users for voting the alternatives based on specified evaluation criteria. These votes are statistically verified by yet another agent in order to find significant similarities among the votes to derive the rankings for the alternatives (Steps 7-8 of the MWAF process).

5 Case study

The GPSE system has been applied to several cases including the followings:

- Selection of Agent-Oriented Software Engineering (AOSE) methodologies
- Selection of software testing tool
- Selection of COTS – single
- Selection (configuration) of COTS – multiple

In this case study we use the GPSE system for deciding what AOSE methodology is the best to adopt for developing a multi-agent system. So far, there is no industry-wide agreement on the kinds of features a methodology should support. Evaluation is a crucial and critical task here to identify the differences between several AOSE methodologies. The GPSE provides a reliable solution with accurate results based on applying state-of-the-art statistical procedures to evaluate AOSE methodologies and comes up with a set of measures that help in selecting the most appropriate methodology for developing prospective agent-based applications.

To select the software items being evaluated (e.g., AOSE methodologies in our case), we started by

conducting a primarily comparative survey to review a large number of AOSE methodologies and select the most qualified ones. After reviewing 31 properly-documented methodologies against the qualification assumptions (Section 3.3 Step 1), the following 9 methodologies were selected: Gaia, MaSE, Tropos, Agent-SE, MASSIVE, Prometheus, MESSAGE, MAS-Common-KADS, and PASSI [10].

Then the dimensions were identified. We studied the selected nine methodologies comprehensively to identify the most important and common measures that will be used as evaluation criteria. Consequently, we came up with six primarily criteria that we indicated by the following dimensions:

- Dimension 1: Agency-related attributes
- Dimension 2: Modeling-related attributes
- Dimension 3: Communication-related attributes
- Dimension 4: Process-related attributes
- Dimension 5: Application-related attributes
- Dimension 6: User-perception attributes

Then the relevant attributes for each dimension were identified. We broke down each dimension into a number of relational attributes that describe its main features as follows:

Dimension 1: Agency-related attributes

This dimension contains attributes that address features involving the internal properties and basic architecture of agents. The hierarchical structure of this dimension is shown in Figure 7.

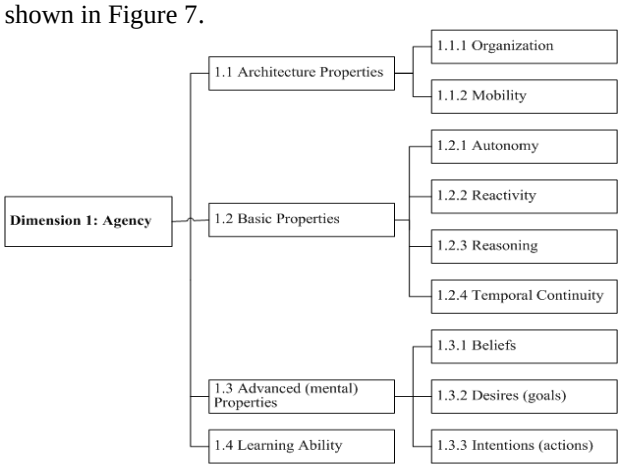

Figure 7: Hierarchical structure of Dimension 1.

Dimension 2: Modeling-related attributes

This dimension consists of the following attributes that address and examine specific features to describe the most common and important aspects to model agents.

1. Notation
2. Expressiveness
3. Abstraction
4. Consistency
5. Concurrency
6. Traceability
7. Derivation and reusability

Dimension 3: Communication-related attributes

This dimension encompasses the following attributes that address features related to the possible interactions and interfacing of agents.

1. Local Communication: cooperation; coordination; competition; negotiation.
2. Wide Communication: interaction with the external environment; agent-based user interface; subsystems interaction.

Dimension 4: Process-related attributes

This dimension encompasses a number of attributes that are given by the following hierarchy to address and examine several issues involving the development process of agents and multiagent systems.

1. Development lifecycle: architectural design; detailed design; verification and validation
2. Refinability
3. Managing complexity

Dimension 5: Application-related attributes

This dimension includes attributes that address and assess different aspects involving the methodology's applicability, and examine some socio-economic factors that affect the decision of recommending and adopting an AOSE methodology.

1. Applicability
2. Maturity
3. Field history
4. Cost concerns

Dimension 6: User perception attributes

In order to make a decision on whether to adopt a specific AOSE methodology, perception, which is entirely a subjective feature, is important and substantial. This is due to the effect of the natural intentionality in human behavior [12]. User perceptions are assessed through the following attributes:

1. Perceived ease of use
2. Perceived usefulness
3. Intention to use

In the next step we used the GPSE system to select the appropriate statistical model and procedure that can fit to our data. In this step we have to determine the proper number of observations needed (and consequently the number of evaluators needed to give their feedback to the evaluation questionnaire) to achieve reasonable accuracy of the statistical analysis. In this case, we have nine methodologies that will be treated statistically as *treatments* and we decided to have at least 4 replicates per treatment. Balanced Incomplete Block Design (BIBD) model was selected.

By denoting the 9 methodologies with letters from A to I and assigning each block of 3 methodologies- after selecting them in such a way to be as homogeneous as possible- to one participant, we can

obtain 36 replicates that will sufficiently satisfy our goal of having 4 replicas per treatment for 12 participants. Table 4 shows this assignment.

The collected data was validated to assure completeness and accuracy using the tests suggested in Table 3. Then the statistical analysis unit was deployed. Followings are the main results of this process.

In order to determine whether significant differences exist among the evaluated methodologies, we conducted separate experiment for each of the six dimensions that characterize the nine methodologies. That is, we will conduct six individual experiments.

Table 4: BIBD tableau for blocks and treatments.

Block (Participants), *j*	Treatments (AOSE Methodologies), *i*								
	M1	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**
	A	**B**	**C**						
				D	**E**	**F**			
							G	**H**	**I**
	A			**D**			**G**		
		B			**E**			**H**	
			C			**F**			**I**
	A					**F**		**H**	
		B		**D**					**I**
			C		**E**		**G**		
	A				**E**				**I**
		B				**F**	**G**		
			C	**D**				**H**	

M = Methodology; P = Participant

In this context, each set of attributes representing a specific criterion given by a dimension were investigated statistically over the nine methodologies. This helped determine whether the strength or effectiveness of this dimension differs among the evaluated methodologies. The following set of hypotheses describes this strategy.

Null hypothesis: H_0: $\tau_i=0$, for i =1 to 9

Indicating that there is **no** significant difference in the mean effectiveness of the examined dimension among the evaluated AOSE methodologies.

Alternative hypothesis, H_a: at least one $\tau_i \neq 0$

Implying that there is significant difference in the mean effectiveness.

The statistical analysis unit in GPSE showed that the mean effectiveness of all the evaluated dimensions (except Dimension 5: Application-related) differs among the evaluated nine methodologies. As a result, the methodologies were ranked for each dimension according to their estimated adjusted mean of effectiveness as shown in Table 5.

Table 5: Ranking evaluated methodologies for each dimension based on mean of effectiveness.

Dimension 1: Agency			
❶	M1: Gaia	$\hat{\mu}_1$ =	6.494
❷	M2: MaSE	$\hat{\mu}_2$ =	5.861
❸	M3: Tropos	$\hat{\mu}_3$ =	5.835
❹	M9: PASSI	$\hat{\mu}_9$ =	5.713
❺	M8: MAS-Common	$\hat{\mu}_8$ =	5.549
❻	M7: MESSAGE	$\hat{\mu}_7$ =	5.524
❼	M4: Agent-SE	$\hat{\mu}_4$ =	5.192
❽	M6: Prometheus	$\hat{\mu}_6$ =	5.049
❾	M5: MASSIVE	$\hat{\mu}_5$ =	4.684

Dimension 2: Modeling			
❶	M2: MaSE	$\hat{\mu}_2$ =	6.593
❷	M9: PASSI	$\hat{\mu}_9$ =	6.428
❸	M1: Gaia	$\hat{\mu}_1$ =	6.037
❹	M7: MESSAGE	$\hat{\mu}_7$ =	5.777
❺	M8: MAS-Common	$\hat{\mu}_8$ =	5.560
❻	M5: MASSIVE	$\hat{\mu}_5$ =	5.271
❼	M6: Prometheus	$\hat{\mu}_6$ =	5.074
❽	M4: Agent-SE	$\hat{\mu}_4$ =	4.755
❾	M3: Tropos	$\hat{\mu}_3$ =	4.580

Finally, the evaluated methodologies were ranked according to the accumulated proportional order of their dimensions. For example, the *Gaia* methodology, M1, has the following accumulated proportional order: 1(←D1) + 7/9(←D2) + 4/9(←D3) + 1(←D4) + 8/9(←D6), where the arrow points to the dimension contributing the proportional value. In this way, we determined the accumulated proportional order of each methodology as well as the overall ranking as shown in Tables 6, 7 and 8. Note that we discarded the proportional orders given to dimension D5 because no significant differences were detected.

Table 6: Dimension ranks for the AOSE methodologies.

Methodology, M_i	**Order**	**Proportional Order**	**Dimension, D_i**					
			D1	**D2**	**D3**	**D4**	**D5**	**D6**
	1	9/9	M1	M2	M7	M1	M2	M2
	2	8/9	M2	M9	M8	M2	M1	M1
	3	7/9	M3	M1	M9	M5	M9	M7
	4	6/9	M9	M7	M2	M8	M7	M8
	5	5/9	M8	M8	M5	M9	M5	M5
	6	4/9	M7	M5	M1	M6	M8	M9
	7	3/9	M4	M6	M3	M4	M3	M4
	8	2/9	M6	M4	M4	M7	M4	M3
	9	1/9	M5	M3	M6	M3	M6	M6

Table 7: Accumulated proportional order of the nine methodologies against the evaluated six dimensions.

Methodology	**Total weight**
M1: Gaia	[9+7+4+9+8]/9 = 37/9
M2: MaSE	[8+9+6+8+9]/9 = 40/9
M3: Tropos	[7+1+3+1+2]/9 = 14/9
M4: Agent-SE	[3+2+2+3+3]/9 = 13/9
M5: MASSIVE	[1+4+5+7+5]/9 = 22/9
M6: Prometheus	[2+3+1+4+1]/9 = 11/9
M7: MESSAGE	[4+6+9+2+7]/9 = 28/9
M8: MAS-Common	[5+5+8+6+6]/9 = 30/9
M9: PASSI	[6+8+7+5+4]/9 = 30/9

Table 8: Overall ranking of the AOSE methodologies.

Rank	1	2	3	3	4	5	6	7	8
Method	M2	M1	M8	M9	M7	M5	M3	M4	M6

6 Conclusions and future works

In the present day market, a customer is faced with various alternatives in the selection and purchase of a software product or deployment of a certain process. This work focused on designing and developing a General Purpose Software Evaluation (GPSE) system. The main objective of the GPSE system is to evaluate various software systems, that are available for a given business application, in order to select the most suitable product or process that meets the requirements of the application as well as the preferences of expert users in an effective manner. In order to obtain expert users' ratings for the products or processes, the Multi-dimensional Weighted Attribute Framework (MWAF) is proposed and adapted [10]. The framework allows the user to define and configure significant evaluation criteria in the form of dimensions and attributes for each dimension. Each treatment considered in the evaluation is rated as per the defined criteria. The GPSE system makes use of statistical analysis based on ANOVA and pairwise comparison tests for ranking the software products or processes. The user of the system is provided with overall ranking of the evaluated systems as well as ranking in all the major evaluation areas. This analysis will help the user in selecting an appropriate product or process to address his/her business needs. This system can be used for the evaluation of any software, hardware or any other product or system where there is a need for the selection to be made from among various alternatives with similar functionalities.

The stand-alone GPSE system, in its current implementation, consists of GUI, database, and a statistical analysis unit. The design of the system offers adequate flexibility to enable users to adapt and use the system for a variety of applications that require complex decision making based on the evaluation of multiple options. Furthermore, the technological choices were made with a due consideration on the portability of the system onto a variety of platforms thus enhancing the overall utility of the concept.

The GPSE system uses ANOVA method for assessing whether there are significant differences among the products being evaluated against a given set of dimensions and their attributes. As a result, if the underlying data does not support the assumptions that need to be satisfied for the application of ANOVA, the method may not be employed effectively. For such cases, usage of suitable data mining techniques will be helpful. Furthermore, the MWAF framework which is the foundation of the present GPSE system requires sufficient prior knowledge on the part of a user to determine the criteria that are considered significant for the evaluation. Eliciting and implementing this knowledge inside the system will contribute to the improved usability of the GPSE system.

References

[1] Alves C., and Castro, J., "CRE: A Systematic Method for COTS Selection", *Proc. XV Brazilian Symposium on Software Engineering*, Rio de Janeiro, Brazil, 2001.

[2] Bayer, P., and Svantesson, M., "Comparison of Agent-oriented Methodologies Analysis and Design," *Programming, Blekinge Institute of Technology (BITSWAP)*, 2001.

[3] Burgues, X., Estay, C., Franch, X., Pastor, J.A., and Quer, C., "Combined Selection of COTS Components", *Proc. 1st Int. Conf. on COTS-Based Software Systems (ICCBSS'02)*, Orlando, Florida, 2002, pp. 54-64.

[4] Cavanaugh, B.P., and Polen, S.M., "Add Decision Analysis to Your COTS Selection Process", *Journal of Defense Software Engineering*, April 2002.

[5] Clarke, G., and Kempson, R., "Introduction to Design and Analysis of Experiments," John Wiley & Sons, Inc., 1997.

[6] Cochran, W., and Cox, G., "Experimental Designs," 2nd ed., John Wiley & Sons, Inc, 1992.

[7] Chung, L., Nixon, B., Yu, E., and Mylopoulos, J., "Nonfunctional Requirements in Software Engineering," Kluwer Academic Press, 2000.

[8] Chung, K., Cooper, K., and Courtney, C., "COTS-Aware Requirements Engineering: The CARE Process", *Proc. 2nd International Workshop on Requirements Engineering for COTS Components*, Kyoto, Japan, September 7, 2004.

[9] Dam, K., and Winikoff, M., "Comparing Agent-Oriented Methodologies," *Proceedings of the 5th Int'l Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS'03)*, Melbourne, Australia, 2003, 78-93.

[10] Elamy, A., "A Statistical Approach for Evaluating Agent-Oriented Software Engineering Methodologies," MSc. thesis, Department of Electrical and Computer Engineering, University of Calgary, 2005.

[11] Elamy, A., and Far, B., "A Multidimensional Weighted-Attributes Framework (MWAF) for Evaluating Agent-Oriented Software Engineering Methodologies," *Proceedings of the 19th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE'06)*, Ottawa, Canada, 2006.

[12] Fishbein, M. , and Ajzen, I., "Belief, Attitude, Intention and Behavior: An Introduction to Theory and Research," Addison-Wesley, Boston, 1975.

[13] Gross, D., and Yu, E., "Evolving System Architecture to Meet Changing Business Goals: An Agent and Goal-Oriented Approach," *Proceedings of the ICSE-2001 Workshop: From Software Requirements to Architectures (STRAW'01)*, 2001, pp. 16-21.

[14] Juneidi, S., and Vouros, G., "Evaluation of Agent Oriented Software Engineering Main Approaches," *Proceedings IASTED Int'l Conf. on Software Engineering (SE'04)*, Innsbruck, Austria, 2004.

[15] Kontio, J., "A Case Study in Applying a Systematic Method for COTS Selection", *Proc. ICSE-18*, 1996, pp. 201–209.

[16] Kunda, D., "A social-technical approach to selecting software supporting COTS-Based Systems", *PhD Thesis*, Department of Computer Science, University of York, Oct. 2001.

[17] Liu, M., and Chan, L., "Uniformity of incomplete block designs," Int'l Journal Materials and Product Technology, vol. 20, no. 1–3, 2004, pp.143–149.

[18] Ncube C., and Maiden, N.A.M., "Guiding Parallel Requirements Acquisition and COTS software", *Proc. IEEE International Symposium on Requirements Engineering*, 7-11 June 1999, pp. 133 – 140.

[19] Neter, J., Wasserman, W., and Kutner, M., "Applied Linear Statistical Models," 5th ed., Irwin, USA, 1996.

[20] Silva, C., Tedesco, P., Castro, J., and Pinto, R., "Comparing Agent-Oriented Methodologies Using NFR Approach," *Proceedings of the 3rd Workshop on Software Engineering for Large-Scale Multi-Agent Systems (SELMAS'04)*, Edinburgh - Scotland, vol 1, 2004, pp. 1-9.

[21] Shehory, O., and Sturm, A. "Evaluation of Modeling Techniques for Agent-Based Systems," *Proceedings of the 5th Int'l Conference on Autonomous Agents*, May 2001, Montréal, pp. 624-631.

[22] Stevens, S., "Handbook of Experimental Psychology," John Wiley and Sons, New York, 1951.

[23] Sudeikat, J., Braubach, L., Pokahr, A., and Lamersdorf, W., "Evaluation of Agent-Oriented Software Methodologies – Examination of the Gap between Modeling and Platform," *Proceedings of the 5th Int'l Workshop on Agent-Oriented Software Engineering (AOSE'04)*, Jul. 2004, pp. 126-141.

[24] Tran, Q., Low, G., and Williams, M., "Comparison of Ten Agent-Oriented Methodologies," Agent-Oriented Methodologies, B. Henderson-Sellers, P. Giorgini, Eds., Idea Group Inc., PA, USA, 2005.

[25] Yates, F., "Experimental Design: Selected Papers," Griffin, UK, 1970.